

Amendment of Claims

Please amend the claims as indicated in the following listing of claims. This listing of claims will replace all prior versions and listings of claims in the present application.

Listing of Claims

1. (original) A method for detecting malicious software within or attacking a computer system, said method comprising the steps of:

in response to a system call, executing a hook routine at a location of said system call to (a) determine a data flow or process requested by said call, (b) determine another data flow or process for data related to that of said call, (c) automatically generate a consolidated information flow diagram showing said data flow or process of said call and said other data flow or process, and after steps (a-c), (d) call a routine to perform said data flow or process requested by said call.

2. (original) A method as set forth in claim 1, wherein a user monitors said information flow diagram and compares the data flow process of steps (a) and (b) with a data flow or process expected by said user.

3. (original) A method as set forth in claim 1, wherein said information flow diagram illustrates locations of said data at stages of a processing activity.

4. (original) A method as set forth in claim 1, wherein said system call is selected from the set of: open file, copy file to memory, copy memory to register, mathematical functions, write to file, and network or communication functions.

5. (original) A method as set forth in claim 1, wherein said system call is a software interrupt of an operating system.

6. (original) A method as set forth in claim 1, wherein said system call causes a processor to stop its current activity and execute said hook routine.

7. (original) A method as set forth in claim 1 wherein said system call is made by malicious software.

8. (original) A system for detecting malicious software in a computer system, said system comprising:

means, responsive to a system call, for executing a hook routine at a location of said system call to (a) determine a data flow or process requested by said call, (b) determine another data flow or process for data related to that of said call, (c) automatically generate a consolidated information flow diagram showing said data flow or process of said call and said other data flow or process, and after steps (a-c), (d) call a routine to perform said data flow or process requested by said call; and

means for displaying said information flow diagram.

9. (original) A system as set forth in claim 8, wherein said information flow diagram illustrates locations of said data at stages of a processing activity.

10. (original) A system as set forth in claim 8, wherein said system call is selected from the set of: open file, copy file to memory, copy memory to register, mathematical functions, write to file, and network or communication functions.

11. (original) A system as set forth in claim 8, wherein said system call is a software interrupt of an operating system.

12. (original) A system as set forth in claim 8, wherein said system call causes a processor to stop its current activity and execute said hook routine.

13. (original) A system as set forth in claim 8 wherein said system call is made by malicious software.

14. (original) A computer program product for detecting malicious software in a computer system, said computer program product comprising:

a computer readable medium;

program instructions, responsive to a system call, for executing a hook routine at a location of said system call to (a) determine a data flow or process requested by said call, (b) determine another data flow or process for data related to that of said call, (c) automatically generate a consolidated information flow diagram showing said data flow or process of said call and said other data flow or process, and after steps (a-c), (d) call a routine to perform said data flow or process requested by said call; and wherein

said program instructions are recorded on said medium.

15. (new) A method as set forth in claim 1, wherein said hook routine is at a location pointed to by said system call.